Behavioral On-Line Advertising

Fabrizio Caruso
Neodata Group, Catania
fabrizio.caruso@neodatagroup.com

Giovanni Giuffrida
Dept. of Social Sciences, University of Catania
ggiuffrida@dmi.unict.it

Calogero Zarba
Neodata Group, Catania
calogero.zarba@neodatagroup.com

January 19, 2011

Abstract

We present a new algorithm for behavioral targeting of banner advertisements. We record different user's actions such as clicks, search queries and page views. We use the collected information on the user to estimate in real time the probability of a click on a banner. A banner is displayed if it either has the highest probability of being clicked or if it is the one that generates the highest average profit.

Keywords: web advertisement, behavioral targeting, association rule, data mining, click-through rate.

1 Introduction

The setting of our problem is the following: we are given a finite set of users, and a webserver. At each instant of time, a user u may connect to the webserver, requesting a webpage. The webserver responds to the request, and inserts into the webpage an appropriate banner containing an advertisement. The user may then click on the banner, or he may not.

We take into account different events: impressions (visualizations), clicks, registrations, page views, keywords in a search queries, etc... An impression event occurs when the webserver responds to a user request for a given webpage, and inserts into the webpage a banner. A click event occurs when the user clicks on a banner. A registration is a voluntary action of the user after a click on the banner such as a purchase of the advertised item or the registration into a site and may have different levels depending on the profit/value of the action for the advertiser. A page view is a simple view of a page. A keyword in a search query is the action of search for a specific word in a search engine embedded in a website. We refer to feature events as to the events that can be used to study the behavior of the users. We do not consider clicks and registrations as features because in our model we are assuming the independence of features (see equation 4 in Section 3 for the technical details). In Section 6 we describe a heuristic improvement of our method that also considers clicks as features, similarly to methods used in collaborative filtering. A good choice could be to take as features all voluntary feature events (except impressions).

Each impression, click or registration (purchase of a product, registration into the advertised site, etc. . .) of a banner b can generate a profit. For sake of simplicity we will primarily consider profits generated by clicks and shortly give a description on how both impressions and registrations (see Section 5) can be taken into consideration.

Our goal is to maximize the profit generated by all clicks or maximize the number of clicks. The former goal is more general than the latter, because if we assume unitary value for all clicks, the total profit equals the total number of clicks. This problem has already been treated in the scientific literature (see [4] where the linear Poisson regression model is used to predict click-through rates and [5] where a metric is introduced to assess the quality of the behavioral targeting). Our proposed approach is simpler than the other similar approaches in the literature, in that it uses the naive Bayesian model. Other approaches are possible such as the ones based on linear programming models ([1], [6], [2]).

For a given user u, we store all events of the user in the cookie maintained by the browser. The cookie contains also the timestamps of each stored event. The user's cookie is used by the webserver each time the user requests a webpage, in order to select an appropriate banner.

In this paper, we describe an algorithm that allows the webserver to select an appropriate banner, based on the information stored in the user's cookie.

We refer to a *feature* as to the presence of a feature event in the user's cookie. We try to estimate the value of the association rule $f_1, \dots, f_n \to b$, i.e., the probability that a user u clicks on b, provided the u has features $f_1, \dots, f_n$.

We use information on the features and the click-through rates. The webserver keeps track in real time of the click-through rates of all banners among users that have the same feature.

Moreover we propose a heuristics to avoid overflooding a user with the same banner based on the impressions in the user's cookie.

The paper is organized as follows: in Section 2 we introduce some notation and definitions; the algorithm for selecting the banner to display is described in Section 3; the heuristics that limits the number of displays of the same banner is described in Section 4; in the last sections different generalizations of the approach are considered (impressions and registrations in Section 5, clicks as features in Section 6, generalizations in terms of locations and time are considered in Section 7).

2 Preliminaries

We call “impression”, the display of an advertisement. We will be using three data-structures to store information on user’s clicks and impressions at real time: a user’s *history*, which depends on the user; a *click matrix* and an *impression matrix*, which are global.

In the following we denote with $B = \{b_1, \ldots, b_n\}$ the set of all banners and by $\mathcal{F}$ the set of all features to be taken into consideration.

Definition 1 (User’s history)**.** *For every user u we define their* history *as the set of triples* $(\mathsf{type}, \mathsf{obj}, \mathsf{time})$ *that describe all events and timestamps of user u, where* type *is the type of event (impression, click, page view, search query, etc...),* obj *is either the clicked banner, the URL of the viewed page or the keyword in search query, and where* time *is the timestamp of the event.*

Definition 2 (User’s profile)**.** *For every user u we define their* profile $\mathcal{P}_u = (\mathcal{F}_u, \mathcal{S}_u, \mathcal{C}_u)$*, where $\mathcal{F}_u \subseteq \mathcal{F}$, $\mathcal{S}_u : B \to \mathbb{N}$ maps each banner to the number of its impressions to user u, and $\mathcal{C}_u : B \to \mathbb{N}$ maps each banner to the number of its clicks by user u.*

Remark 1. *The user’s history is the only data-structure that needs to be stored in the user’s cookie. We have introduced the user’s profile for the sake of simplicity.*

Definition 3. *We denote by $S = (S_{i,j})$ the* impression matrix*, where $S_{i,j}$ is the number of impressions of banner b_j among users u that have feature i, i.e., $i \in \mathcal{F}_u$.*

Definition 4. *We denote by $C = (c_{i,j})$ the* click matrix*, where $c_{i,j}$ is the number of clicks of banner b_j among users u who have feature i, i.e., $i \in \mathcal{F}_u$.*

3 The banner selection algorithm

Assume that a user u requests a webpage from the webserver, which responds by sending the webpage, and inserting into the webpage an appropriate banner. We now describe the general strategy on how the banner is selected, based on the information stored in the user’s history of u. We will denote by $P(b)$ the global probability for a banner b to be clicked and by $P(b \mid f_1, \ldots, f_n)$ the probability for a banner b to be clicked by a user with features $f_1, \ldots, f_n$. We are also assuming that $P(b) \neq 0$, $P(b \mid f_i) \neq 0$, $\forall i$. In particular we want to maximize the probability $P(b \mid f_1, \ldots, f_n)$.

Given a set K of candidate banners that may be selected by the webserver, for each $b \in K$ we compute a score $\mathit{score}(b)$. The banner with the highest score is then selected by the webserver.

As score we take

$$\mathit{score}(b) = \mathit{cpc}(b) \cdot \mathit{rule}(f_1, \ldots, f_n \to b); \tag{1}$$

where

- $cpc(b)$ is the "cost per click" (the profit) of b;
- $rule(f_1, \ldots, f_n \to b)$ is defined as follows:

$$rule(f_1, \ldots, f_n \to b) = P(b) \prod_{i=1}^{n} \frac{P(b \mid f_i)}{P(b)}. \tag{2}$$

Given two expressions α and β we use use the notation $\alpha \propto \beta$ to mean $\alpha = c\beta$ where c does not depend on b.

In particular we have the following fact

Fact 1. *Under the hypothesis that the features f_i are independent events we have*

$$rule(f_1, \ldots, f_n \to b) \propto P(b \mid f_1, \ldots, f_n). \tag{3}$$

Proof. By applying Bayes' Theorem twice, under the simplifying hypothesis of independent features, we have

$$\begin{aligned}
P(b \mid f_1, \ldots, f_n) &= \frac{P(b)P(f_1, \ldots f_n \mid b)}{P(f_1, \ldots, f_n)} \propto P(b)P(f_1, \ldots f_n \mid b) \\
&= P(b) \prod_{i=1}^{n} P(f_i \mid b) = P(b) \prod_{i=1}^{n} \frac{P(b \mid f_i)P(f_i)}{P(b)} \\
&= P(b) \prod_{i=1}^{n} \frac{P(b \mid f_i)}{P(b)} \prod_{i=1}^{n} P(f_i) \propto P(b) \prod_{i=1}^{n} \frac{P(b \mid f_i)}{P(b)} \\
&= rule(f_1, \ldots, f_n \to b)\,.
\end{aligned} \tag{4}$$

□

Therefore the banner with highest *rule* is the banner with the highest probability of been clicked and the banner with highest score is the banner that generates the highest average profit per click.

3.1 Click-through rates

In order to compute the probabilities $P(b)$, $P(b \mid f_i)$ for $i = 1, \ldots, b.$ in (4) we use the concept of click-through rate:

$$\begin{aligned}
ctr(b) &= \frac{\#\text{clicks on } b}{\#\text{impressions of } b} \text{ (among all users)}. \\
ctr_f(b) &= \frac{\#\text{clicks on } b}{\#\text{impressions of } b} \text{ (among users with feature } f\text{)}.
\end{aligned} \tag{5}$$

Therefore we can compute the probabilities $P(b)$, $P(b \mid f_i)$ as click frequencies. Hence we have

$$P(b) = ctr(b); \;\; P(b \mid f_i) = ctr_{f_i}(b). \tag{6}$$

Thus we can write

$$\mathit{rule}(f_1, \ldots, f_n \to b) = \mathit{ctr}(b) \cdot \prod_{i=1}^{n} \frac{\mathit{ctr}_{f_i}(b)}{\mathit{ctr}(b)} = \mathit{ctr}(b)^{1-n} \prod_{i=1}^{n} \mathit{ctr}_{f_i}(b). \tag{7}$$

3.2 Updating relative click-through rates

In order to keep up to date the click-through rates for each feature we need to update the impression matrix and the click matrix in real time. A click-through rate of banner b_j for a given feature i is then computed by counting the clicks and impressions in the i-th rows of the two matrices:

$$\mathit{ctr}_i(b_j) = \frac{c_{i,j}}{s_{i,j}}. \tag{8}$$

We consider the set $B = \{b_1, \ldots, b_n\}$ of all banners. In order to update the matrices after each impression and click, for every user, first, the user's history (and profile) are updated, and second, the following actions on the matrices are taken:

- If there is an impression on banner b_j by user u: then for every feature i of u, $s_{i,j}$ is increased by one: for every i such that $i \in \mathcal{F}_u$ we do $s_{i,j} := s_{i,j} + 1$.
- If there is a click on banner b_j by a user u that has already clicked on b_j, i.e. $\mathcal{C}_u(b_j) > 0$: then for every feature i of u, $c_{i,j}$ is increased by one, i.e. for every i such that $i \in \mathcal{F}_u$ we do $c_{i,j} := c_{i,j} + 1$.
- If there is a feature event i by a user u that did not have that feature: the $i - th$ rows in C and S are updated with respectively the impressions and the clicks in the user's profile: $s_{i,j} := s_{i,j} + \mathcal{S}_u(b_j)$, $c_{i,j} := c_{i,j} + \mathcal{C}_u(b_j)$, $\forall j$.

4 Avoiding user's boredom

In this section we describe our strategy on how to avoid overflooding a user with the same banner. We achieve this by "throttling down" the value of the candidate banner taking into account the times at which the banner has already been displayed to the user. The value $\mathit{rule}(f_1, \ldots, f_n \to b)$ is multiplied by a scaling factor $\mathit{throttle}_u(b)$ with the following properties:

I. $0 < \mathit{throttle}(b) \leq 1$: in order to have a down-scaling.

II. $\mathit{throttle}(b)$ decreases with the number of impressions of b.

III. $\mathit{throttle}(b)$ decreases more if the impressions are more recent and it increases if the impressions are farther in the past.

So that we have

$$\mathit{score}(b) = \mathit{cpc}(b)\, \mathit{rule}(f_1, \ldots, f_n \to b)\, \mathit{throttle}(b)\,.$$

In particular we choose the following function. Let t be the current instant of time. Also, let t_i, for $i = 1, \ldots, m$ be the instants of time in which an impression event for banner b and user u has occurred.

$$\mathit{throttle}(b) = \prod_{i=1}^{m} \left(1 - \alpha \left(\frac{1}{2} \right)^{(t-t_i)/h} \right) \,, \tag{9}$$

where $0 < \alpha < 1$ and h are heuristically selected parameters.

Thus the formula in (9) can avoid overflooding the user with the same banners and can improve the estimation of the probability of a click. Therefore we have the following facts

- $\mathit{rule}(f_1, \ldots, f_n \to b)\, \mathit{throttle}(b)$ is an approximation of the probability of a click on banner b by a user who has features $f_1, \ldots, f_n$ and has already seen certain impressions of banner b.
- $\mathit{score}(b)$ is an approximation of the expected average profit solely generated by a possible click after an impression of banner b.

5 Impressions and Registrations

In the most general case we may have banners that generate a profit for each impression, click and registration.

For each candidate banner b, we can take into account profits generated by both impressions and clicks by considering:

$$\mathit{score}^{+}(b) = \mathit{imp\_profit}(b) + \mathit{score}(b); \tag{10}$$

where $\mathit{imp\_profit}(b)$ is the profit generated by the display of b.

This approach can be further generalized to encompass registrations of any step by simply treating them as click-like events.

6 Click events as features

Our approach only considers non-click events as features and assumes that the features are independent of each other. We can improve the accuracy of our predictions by considering click events but they would have to be treated differently because the basic assumption of independence does not hold for them. Clicks could be treated similarly to a purchase in a collaborative filtering approach. We record each unique click in a user × banner matrix. The probability $P(b \mid c)$ of a click on banner b by a user that has clicked on c will then depend on whether others users that have clicked on c have also clicked on b. For more details on an practical use of this approach we refer to [3], where the approach is used for a recommending system.

7 Space and time

This approach can be generalized in terms of time and space i.e., location of a banner.

We can also take into account these significant attributes in order to better target the users at specific times for banners at specific locations. This can be achieved by simply recording this data in an extra dimension in the matrices S, C described in Section 3.2.

8 Future work

This approach could be further developed, improved and generalized in different respects: with respect to how the features are treated (by considering clusters of features instead of single features or by considering non-Boolean features), and with respect to its applications.

8.1 Non-Boolean features

We can assign each feature a counter that could be used in an extended definition of value of $\mathit{rule}(f_1, \ldots, f_n \to b)$. One possibility could be to consider

$$\mathit{rule}^*(f_1, \ldots, f_n \to b) := W(c_1, \ldots, c_n) \cdot \mathit{rule}(f_1, \ldots, f_n \to b), \tag{11}$$

where

- c_i is the (possibly normalized with respect with the average) counter of feature f_i, for $i = 1, \ldots, n$;
- $W(c_1, \ldots, c_n)$ is a measure of how the counters should correct the association rule $f_1, \ldots, f_n \to b$.

  A possible straightforward candidate for $W(c_1, \ldots, c_n)$ could be the simple arithmetic average:

  $$W(c_1, \ldots, c_n) := \frac{c_1 + \cdots + c_n}{n}. \tag{12}$$

This could be used to differentiate between a single page view (or a single search query with a keyword) and multiple page views (multiple search queries with the same keyword).

8.2 Application to on-line newspapers and magazines

This approach could also be applied to on-line newspapers and magazines in that the visualization of an article's title is seen as an impression and a click on the title is seen as a click on a banner.

References

[1] Naoki Abe and Atsuyoshi Nakamura. Learning to Optimally Schedule Internet Banner Advertisements. In *ICML '99: Proceedings of the Sixteenth International Conference on Machine Learning*, pages 12–21, San Francisco, CA, USA, 1999. Morgan Kaufmann Publishers Inc.

[2] Fabrizio Caruso and Giovanni Giuffrida. Optimizing On-Line Advertising. 2010. Preprint available at `arXiv:1011.5364`.

[3] Fabrizio Caruso, Giovanni Giuffrida, and Calogero Zarba. Subjective Collaborative Filtering. 2010. Preprint available at `arXiv:1101.2834`.

[4] Ye Chen, Dmitry Pavlov, and John F. Canny. Large-scale behavioral targeting. In *KDD '09: Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 209–218, New York, NY, USA, 2009. ACM.

[5] Yan Jun, Liu Ning, Wang Gang, Zhang Wen, Jiang Yun, and Chen Zheng. How much can behavioral targeting help online advertising? In *18th International World Wide Web Conference (WWW2009)*, April 2009.

[6] Atsuyoshi Nakamura. Improvements in practical aspects of optimally scheduling web advertising. In *WWW '02: Proceedings of the 11th international conference on World Wide Web*, pages 536–541, New York, NY, USA, 2002. ACM.